



**Repositorio Institucional de la Universidad Autónoma de Madrid**

<https://repositorio.uam.es>

Esta es la **versión de autor** del artículo publicado en:

This is an **author produced version** of a paper published in:

ACM Transactions on Information Systems 29.2 (2011): 11

**DOI:** <http://dx.doi.org/10.1145/1961209.1961214>

**Copyright:** © 2011 ACM

El acceso a la versión del editor puede requerir la suscripción del recurso

Access to the published version may require subscription

# Effects of Usage based Feedback on Video Retrieval: a Simulation based Study<sup>1</sup>

DAVID VALLET

Universidad Autónoma de Madrid, Spain

FRANK HOPFGARTNER

University of Glasgow, UK

JOEMON M. JOSE

University of Glasgow, UK

and

PABLO CASTELLS

Universidad Autónoma de Madrid, Spain

---

We present a model for exploiting community based usage information for video retrieval, where implicit usage information from past users is exploited in order to provide enhanced assistance in video retrieval tasks, and alleviate the effects of the semantic gap problem. We propose a graph-based model for all types of implicit and explicit feedback, in which the relevant usage information is represented. Our model is designed to capture the complex interactions of a user with an interactive video retrieval system, including the representation of sequences of user-system interaction during a search session. Building upon this model, four recommendation strategies are defined and evaluated. An evaluation strategy is proposed based on simulated user actions, which enables the evaluation of our recommendation strategies over a usage information pool obtained from 24 users performing four different TRECVID tasks. Furthermore, the proposed simulation approach is used to simulate usage information pools with different characteristics, with which the recommendation approaches are further evaluated on a larger set of tasks, and their performance is studied with respect to the scalability and quality of the available implicit information.

Categories and Subject Descriptors: H.5.1 [Information Interfaces and Presentation] - Multimedia Information Systems - *Evaluation/methodology; Video*; H3.3 [Information Storage and Retrieval] - Information Search and Retrieval - *Search process; Information filtering*

General Terms: Algorithms, Design, Experimentation, Human Factors

Additional Key Words and Phrases: Human-computer interaction, Evaluation model, Collaborative filtering, Implicit feedback

---

## 1. INTRODUCTION

In recent years, multimedia content available to users has increased exponentially. This phenomenon has come along with (and to a great extent is the consequence of) a rapid development of tools, devices, and social services which facilitate users the creation, storage and sharing of personal multimedia content. A new landscape for business and innovation opportunities in multimedia content and technologies has naturally emerged

---

<sup>1</sup> This paper is based on previous work presented at the 30<sup>th</sup> European Conference on Information Retrieval (ECIR 2008) [Vallet et al. 2008]. There are several extensions presented in this study, such as a revision of the usage information representation model, new variations of the recommendation approaches, and a novel methodology to simulate a usage based information pool, to name a few. In addition, there is a more comprehensive evaluation, with an extended test bed and the investigation on the effect of the quality and quantity of the available usage information.

from this evolution, at the same time that new problems and challenges arise. In particular, the hype around social services dealing with visual content, such as Flickr or YouTube, has led to a rather uncoordinated publishing of video data by users worldwide [Cunningham and Nichols 2008]. Furthermore, existing tools to organise and retrieve video content are insufficient in all terms (effectiveness, efficiency and usefulness). Hence, there still is a growing need to develop new retrieval methods that support the users in searching and finding videos they are interested in.

In this context, the semantic gap problem [Smeulders et al. 2000], which is the missing association between the data representation based on the low-level features and the high-level concepts users associate with video, has been identified as a major challenge that many research efforts in the area are addressing. Yet at the time of this writing, the semantic gap is still considered the most important open issue to overcome in multimedia retrieval. A major trend explored in Multimedia Information Retrieval (MIR) to address the semantic gap problem is the adoption of interactive retrieval approaches, where the system further elaborates on information need formulations provided by users, and enhances them by getting additional feedback from the users [Urban et al. 2006a].

This paper delves into the development of effective interactive multimedia retrieval systems by leveraging implicit user feedback from large user communities. We investigate a collaborative approach to overcome the limitations of implicit evidence as available in multimedia retrieval systems, by combining individual and global user feedback and building rich structures from collective user interaction with the system. In order to address the inherent difficulty of formal Interactive Information Retrieval (IIR) evaluation, we propose a simulative evaluation framework enabling the reproduction, variation and upscaling of user-based experiments, using moderate-cost user studies as seed for the simulations.

### 1.1 Interactive Multimedia Retrieval

In a classic information retrieval (IR) scenario, a user formulates a search query and triggers a retrieval process which results in a list of ranked documents in decreasing order of relevance. Various ranking approaches have been introduced based on theories that model whether a document is relevant to a given query or otherwise [Salton 1988]. All approaches however face a major problem: an average search query consists of 2 or 3 query terms [Jansen et al. 2000, Arampatzis and Kamps 2008], resulting in a rather unspecific expression of the user's information need. Without further knowledge, it is hence a demanding task to adapt retrieval results to the user. IIR aims to improve the classic IR paradigms by studying how to further engage users in the retrieval process, in a way that the system can have a more complete understanding of the user's information need.

Most IIR approaches are based on relevance feedback techniques [Salton and Buckley 1990]. The general principle underlying relevance feedback is that the retrieval system is able to enrich an initial user query by identifying the set of relevant documents that were initially returned. Two types of relevance feedback can be broadly distinguished in IIR: implicit and explicit. Explicit relevance feedback requires the user to actively mark relevant documents in the result set, whereas implicit relevance feedback requires the retrieval system to guess which documents were relevant to the user, by interpreting the user interactions with the successive result lists. Although explicit relevance feedback is more reliable, it has been shown in text retrieval that giving explicit relevance feedback is a cognitively demanding task and can affect the search process [Nichols 1997]. This is why implicit feedback is currently the dominant approach in interactive text retrieval [White 2004; Joachims et al. 2007]. However, implicit feedback has been as of yet scarcely explored in interactive video retrieval [Hopfgartner et al. 2008].

In the text retrieval domain, the relevance feedback process usually involves modifying the user query with the information of the most frequent terms of relevant rated documents. With multimedia content however, exploiting relevance feedback becomes trickier by the lack of textual annotation. For instance, YouTube allows submitters to associate text descriptions and tags to uploaded videos, but specific shots or parts of a video cannot be annotated. Other video corpora, such as private collections, are often not annotated, wherefore extracting terms from relevant rated documents is not a feasible approach. One way to bypass the lack of text is to exploit low-level features and use them as a query expansion source, but, as the evaluation runs of the TREC Video Retrieval Evaluation (TRECVID) show, query enrichment by low-level features is not competitive with equivalent text-based expansion methods [Smeaton 2006]. The limitations of implicit feedback have also been addressed by means of collaborative strategies, which provide document recommendations to the target user based on the relevance feedback of past users who undertook similar search tasks (e.g. who input similar queries, accessed similar documents, etc.). Collaborative strategies have been successfully applied to enhance implicit relevance feedback in traditional text search (e.g., [Sun et al. 2005; White et al. 2007]). Collaborative filtering has been a major approach as well in content-blind (thus suitable for multimedia content) recommender systems [Adomavicius and Tuzhilin 2005]. However, the use of collaborative implicit feedback to better answer queries in MIR systems has not been investigated in depth.

## 1.2 Researched approach

In this work we propose an IIR approach for video retrieval systems that addresses the above limitations by leveraging feedback information from large communities of users. The main principles of our approach are as follows. First, in order not to burden the user with additional effort, we take an implicit relevance feedback approach, based on usage information internally collected by the retrieval system while the user is engaged in a search session. Second, in order to cope with the lack of textual annotation in video collections, we investigate collaborative recommendation strategies based on graph structures for the representation, storage and exploitation of usage information from past users. The recommendation approaches build upon a pool of information implicitly collected from users, henceforth denoted as the *implicit pool*. The graph structures built from this pool induce a collaborative model of user behaviour, collating implicit relevance feedback from a large group of past users. This graph-based model takes hold of the variety and complexity of the interactions that are possible in an interactive MIR system, which is taken as a basis for producing recommendations in a retrieval session.

Four recommendation strategies are explored in this paper: two state of the art algorithms, adapted to our framework, and two new recommendation methods, proposed here. Taking into consideration that these algorithms in turn admit a number of variants, an experimental methodology is needed that facilitates the evaluation of multiple recommendation algorithms as part of the same experiment. Although part of the data used in our evaluation was collected in a user study, the cost of evaluating all the proposed recommendation approaches through a fully user-based study can easily raise feasibility issues. Moreover, our evaluation requirements are such that the experiments should be easily repeatable, in order to study the effect of different variables within a reasonable amount of experimentation time. In order to address these concerns, we introduce a simulation-based methodology, in which implicit relevance feedback techniques can be analysed by simulating user actions over a real video retrieval system [Hopfgartner et al. 2008]. A comparative evaluation of the overall performance of the presented recommendation approaches is provided based on this methodological scheme.

In addition, we are interested in further investigating the impact that different pools of implicit information have on the performance of the proposed methods. We therefore pursue an additional experiment, focusing on two main variables that affect a usage-based collaborative recommendation approach: the size and the quality of the available implicit pool. The size of the implicit pool indicates how many interactions have been previously collected by the system from past users. The quality of the implicit information indicates how close is the implicit feedback to real explicit feedback information, i.e. how trustworthy is the recorded implicit information.

Our analysis on the interaction of users with our video retrieval systems suggests that the quality of implicit feedback obtained from a video retrieval system tends to be much lower compared to the quality of implicit feedback reported for textual [Joachims et al. 2007] and image retrieval systems [Craswell and Szummer, 2007] (see section 6.2.1 for the complete analysis). Thus, a well performing video retrieval recommendation algorithm should be robust to low quality implicit information. In order to assess the effect of this variable, we exploit our simulation framework as a tool to simulate pools of implicit feedback information for different values of pool quality and size. The goal of this evaluation is to enable a fine analysis of the evolution of recommendation performance with respect to the two aforementioned variables, and confirm the hypothesis that performance increases with implicit feedback quality and size. To the best of our knowledge, this is the first time the relation between these variables has been formally investigated.

The remainder of the paper is structured as follows. A summary of related work on implicit feedback applied to MIR and simulation-based evaluation strategies is presented in Section 2. Section 3 will give a brief description of the video retrieval system on which the presented simulation and recommendation models are based. Section 4 introduces the graph-based implicit pool representation, along with the four recommendation strategies. In Section 5, we describe the simulation based evaluation methodology. Section 6 reports and analyses the experimental results. The paper closes in Section 7 with a summary and discussion of the presented work, and concluding remarks.

## 2. RELATED WORK

### 2.1 Implicit Feedback in Multimedia Information Retrieval

Implicit feedback techniques have been successfully applied to retrieval systems in the past. For instance, White [2004] and Joachims et al. [2007] defined and evaluated several implicit feedback models on a text-based retrieval system. Their results indicate that their implicit models were able to obtain a comparable performance to that obtained with explicit feedback models. However, their techniques are based on textual information, and applied individually at runtime during the user's search session. As stated previously, the lack of textual annotation in online video repositories prevents the adoption of this approach in MIR systems. One solution to this problem is to collect the implicit information from a large number of past users, following a collaborative recommendation strategy.

The exploitation of usage information from a community of past users is a widely researched approach to improve IR systems. However, most of past and present studies focus on textual content, and there are only few studies on image [Craswell and Szummer 2007] and video retrieval [Vallet et al. 2008; Hopfgartner et al. 2008]. The main assumption in such systems is that when a user enters a query, the system can exploit the behaviour of past users that were performing similar search sessions [Bauer and Leake 2001; Craswell and Szummer 2007; White et al. 2007]. For instance, Bauer and Leake [2001] build up a task representation based on the user's sequence of accessed

documents. This search task representation is used by an information agent, which proactively suggests documents to the user. Compared to this, our approach builds a more elaborate and complex representation, by processing a more complete and diverse sequence of user interactions, and considering further user action types, other than opening documents.

A commonly exploited usage information structure is the so-called clickthrough data [Craswell and Szummer 2007; Dou et al. 2005; Sun et al. 2005; White et al. 2007]. Clickthrough data is generally limited to the queries entered by users into the system, the documents returned in response, and the documents that the user subsequently opened. Sun et al. [2005] and Dou et al. [2007] mine query log clickthrough information to perform a collaborative personalisation of search results, giving preference to documents that similar users had clicked previously for similar queries. Sun et al. [2005] apply a dimensional reduction pre-processing step on the clickthrough data in order to find latent semantic links between users, queries and documents. Dou et al. [2007] complement these latent relationships with user-topic and document-topic similarity measures. Craswell and Szummer [2007] use a bipartite graph to represent the clickthrough data of an image retrieval system, where queries and documents are the nodes and links are the ones directly captured in the clickthrough data. A random walk is then applied to recommend images based on the user's last query. Similarly to these works, our research is concerned with the exploitation of past usage information to enhance the relevance of system outputs along interactive retrieval sessions. Beyond such works, we research the integration of further sources of implicit relevancy into a feedback pool, besides clickthrough data, such as viewing time, browsing keyframes, or navigating through a video, as proposed in Hopfgartner and Jose [2007]. This additional data enables a richer representation of user actions and potentially better recommendations.

In this line of works, White et al. [2007] explore the concept of query and search session trails, where the interaction between the user and the retrieval system is seen as a trail that leads from the initial user query to the last accessed document of the query session or the search session, the latter consisting of multiple query sessions. White et al. argue that the user's information need is most likely satisfied by the last document of these trails, i.e. the last document that the user accessed in the query or search session. White et al.'s search trail concept is introduced in one of the four recommendation approaches that we evaluate here. While White et al. exploit only the last document of search trails, we are interested in representing and leveraging the whole interaction process. We thus introduce the concept of interaction sequence, which follows our hypothesis that in video retrieval the whole sequence of user actions contains relevant implicit evidence than can be exploited in a collaborative recommendation approach.

## 2.2 Simulation-based Evaluation Methodologies

In the de facto standard IR evaluation methodology known as the Cranfield paradigm, users interact with a system searching for given search topics in a limited dataset. An analysis of recorded transaction log files and the retrieval results is then used to evaluate the research hypothesis [Cleverdon et al. 1966]. One of the main known problems of this approach is the high cost of manually producing usage and relevance data in the required amount to support statistical significance in the results and observations. This becomes particularly critical in interactive IR systems, where the amount of required test variables and experimental dimensions soon make the approach virtually impractical. Simulated interactions have been proposed as an alternative approach to gather the required user feedback. In a simulated approach, a prefixed set of possible steps is considered when a user is progressing through a given search task in the evaluated system [Hopfgartner et al. 2007; Hopfgartner and Jose 2007; White et al. 2004].

In a typical simulation-based evaluation methodology, actions that a real user may do are artificially triggered and used to influence the subsequent retrieval results. Finin [1989] introduced one of the first simulation-based user modelling approaches. His “*General User Modelling System*” (GUMS) allowed software developers to test their systems by feeding them with simple stereotyped user behaviour. White et al. [2004] also proposed a simulation-based approach to evaluate the performance of implicit indicators in textual information retrieval. They simulated user actions consisting of viewing relevant documents, which were exploited by different implicit feedback techniques to improve the retrieval effectiveness. Foley and Smeaton [2008] simulate a synchronous collaborative information retrieval environment, where they exploit log files of independent search sessions by synchronising the start time of different sessions, and analysing how different events in the log files can influence the retrieval process. Hideo et al. [2009] also simulate a synchronous collaborative search scenario, with the purpose of evaluating a number of IR techniques applied to collaborative search. Hopfgartner et al. [2007] introduce a simulation framework to evaluate adaptive multimedia retrieval systems, where implicit user relevance feedback is simulated to evaluate an interactive retrieval system. Hopfgartner and Jose [2007] extended this framework and simulated users interacting with state-of-the-art video retrieval systems. They argue that a simulation can be seen as a pre-implementation method which gives further opportunity to develop appropriate systems and subsequent user-centred evaluations.

We here present an evaluation methodology that takes many ideas from the above work, in particular that a user can be simulated in order to evaluate IIR systems. However, in the context of recommendation techniques, we extend previous work by presenting a simulation methodology that is not only used to evaluate and compare different recommendation approaches, but also to simulate different pools of usage information upon which the approaches are built. This way, an IIR system can be easily tested on usage information displaying different characteristics, and the system’s sensitivity to dataset variables can be studied without an extra manual usage data gathering cost.

### 3. AN INTERACTIVE VIDEO RETRIEVAL SYSTEM

The experimental work in our study leans upon a video retrieval system [Hopfgartner et al. 2008] that we briefly introduce here. As will be explained later, part of the simulation data is obtained from interactions of real users of this system, therefore the usage model instantiation in our study is based on the different types of interactions that this system supports. Note that this particular system does not introduce particular assumptions which would prevent the models and recommendation techniques presented in this paper from being applied to any other retrieval system.

The video retrieval system consists of four main components: a search interface (see Figure 1), a keyframe index, a retrieval engine and our recommendation model. The keyframes in our experiments were indexed based on automatic speech recognition transcripts and machine translation output, which was part of the TRECVID 2006 video collection [Over and Ianeva 2006]. The retrieval engine uses the Okapi BM25 retrieval method, by which retrieval results are ranked in response to keyword-based user searches. In addition to the ranked list of search results, the system provides users with additional recommendations of video shots that might match their search criteria based on a recommendation approach developed for the system, which is a combination of a number of approaches presented in this paper (see [Hopfgartner et al. 2008]).

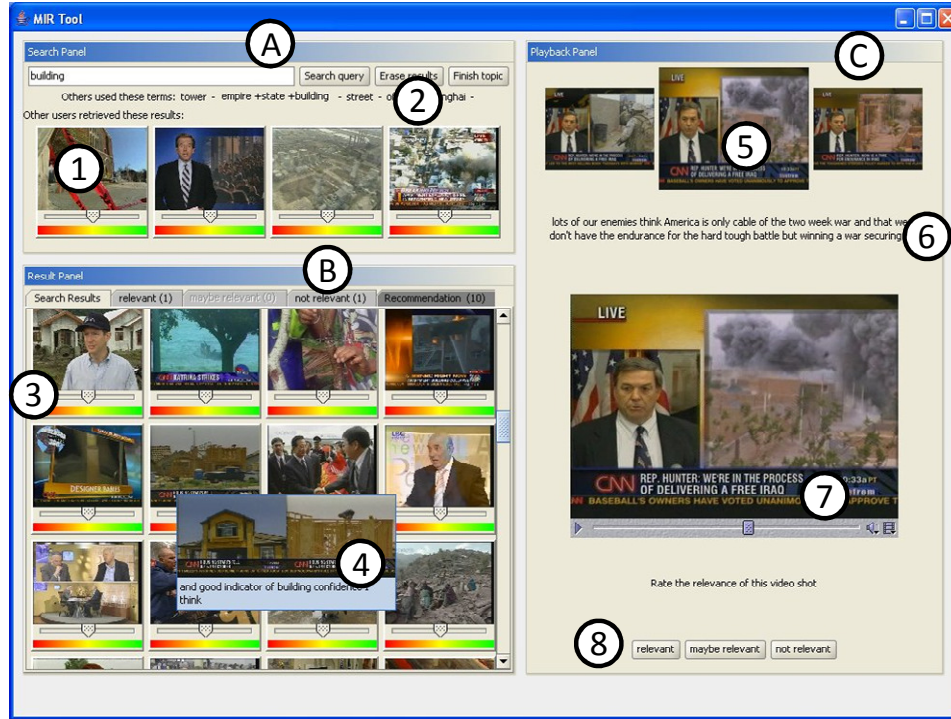


Fig. 1. User interface to the video retrieval system

The user interface of the system consists of three main panels (see Figure 1), search panel (A), result panel (B) and playback panel (C). Users formulate and carry out their searches in the search panel (A), where the system provides interactive assistance for query formulation. As the user enters a text based query, the system recommends alternative search query keywords that can be used to enhance the search (2). The users are also presented with recommendations of video shots that might match their current search process (1), following an initial formulation of the recommendation model presented in this paper.

Search results consist of video clips, ranked by the underlying MIR system. They are displayed in the result panel (B), showing a representative keyframe for each clip. In fact, the panel is divided into five tabs: the results for the current search, a list of results that the user has marked as relevant in the current session, a list of results that the user has marked as perhaps relevant, a list of results that the user has marked as irrelevant, and the list of recommendations that the user has been presented during the search session. Users can mark results in these tabs with a graded relevance value using a sliding bar (3). Additional information about each video shot can be retrieved in the result panel. The displayed video keyframes are featured with a tooltip functionality (4) that shows any text associated with a keyframe when the mouse tip is hovered over it, at the same time that its adjoining keyframes are presented. The playback panel (C) is for viewing video clips (7). As a video is playing, the current keyframe for that clip (5), any text associated with that keyframe (6), and the neighbouring keyframes, are shown in this panel. Users can play, pause, stop and navigate through the video as on a common media player, and also enter explicit relevance judgements on the keyframe (8).

These UI functionalities allow the system to capture and store rich explicit and implicit user feedback, for later exploitation in the recommendation technique. As



described above, explicit feedback is given by users by marking video shots as being either relevant or irrelevant (3, 8). Implicit feedback is produced when users play a video (7); tooltip a video frame (4); or navigate through a video, either by browsing to a neighbour keyframe (5), or by using the navigation bar (7).

The described MIR system is the test-bed for the development and evaluation of the methods presented herein. In consequence, the implicit information representations described next consider the types of implicit actions available in this system (play, tooltip, navigation and video browse). Since our study focuses on implicit relevance techniques, the recorded explicit feedback actions were not used in our experiments. Nonetheless, no particular assumptions are made which would prevent further types of implicit and explicit feedback to be readily included and exploited in the proposed approach.

#### 4. IMPLICIT GRAPH FOR RECOMMENDATION

There are two main desired properties for the usage information representation model in order to implement our recommendation techniques based on users' past usage information. The first property is the representation of all of the user interactions with the system, including the search trails for each interaction. This allows us to fully exploit all of the interactions to provide richer recommendations. The second property is the aggregation of implicit information from multiple sessions and users into a single representation, thus facilitating the analysis and exploitation of past implicit evidence. To achieve these properties we opt for a graph based representation of the users' implicit information.

##### 4.1 Implicit Graph Representation

As introduced earlier, usage information is processed in our approach in the form of graph structures. Two main graph representations are handled in our approach: the first one, a Labelled Directed Multigraph (LDM), gives a full detailed representation of the implicit information, and the second, a Weighted Directed Graph (WDG), is a simplified elaboration of the LDM. It is on top of the WDG where the different recommendation strategies will be defined. Note that the WDG is not dependent on the LDM, it can be derived from it, but it could also be computed directly from raw usage data. Both representations are constructed by means of an aggregation of search session representations. We adopt a definition of a search session that is common in the field: a sequence of user interactions in which time intervals between requests are less than 15 minutes. We formally define LDM and WDG in the remainder of this section, followed by a description of their construction.

*4.1.1 Labelled Directed Multigraph Representation.* In our approach, a user session  $s$  is represented as a set of queries  $Q_s$ , which were input by the user, and the set of multimedia documents  $D_s$  the user accessed during the session. Queries and documents are thus the nodes  $N_s = Q_s \cup D_s$  of our graph representation  $G_s = (N_s, A_s)$ , in which the arcs  $A_s$  are the set of user actions, which we represent in the form  $(n_i, n_j, a, u, t)$ , with  $n_i, n_j \in N_s$  (i.e. either queries or documents), indicating that, at time  $t$ , the user  $u$  performed an action of type  $a$  that led the user from node  $n_i$  to node  $n_j$ . This means that the graph models interactions such as which document the user opened after issuing a query, or the next user action after opening a document, e.g. navigating within the document, opening a related document, etc. Note that  $n_j$  is the object of the action and actions can be reflexive, for instance when a user clicks to view a video and then navigates through it. Action types depend on the kind of actions recorded by the implicit feedback system. For instance, in our video retrieval system (see Section 3) we

considered the following action types: selecting a video, playing a video, navigating through a video, and highlighting a video to view additional metadata [Hopfgartner et al. 2008]. Arcs can contain extra associated metadata as type specific attributes, such as the length of viewing time in a play type action. Additionally, arcs can be weighted following the methodology described in Section 4.1.2. The graph is multilinked, as different actions may have the same source and destination nodes.

The session graph  $G_s = (N_s, A_s)$  is built at runtime based on all the accessed nodes and user actions, capturing the whole interaction process for the user's session  $s$ . Finally, all the session graphs are aggregated into a single graph  $G = (N, A)$ , where  $N = \bigcup_s N_s$  and  $A = \bigcup_s A_s$ , which constitutes a global pool of implicit information. In this step, all of the nodes from the individual graphs are mapped to one large multi-linked graph, where all of the action edges are mapped onto the same graph. The pool graph may not be fully connected, as it is possible that users e.g. selected different paths through the data, or entered a query and took no further actions, etc. Figure 2 a) shows an example of LDM graph.

While the LDM gives a detailed representation of the user interaction with the collection, it is extremely difficult to use in order to provide recommendations. The multiple arcs make the graph extremely complex, which complicates the calculation of the implicit relevance of a node within a single logged session, or for all historical sessions in which the node appears. This motivates the aggregative simplification of LDMs into a simpler representation, based on WDGs.

**4.1.2 Weighted Directed Graph Representation.** In order to exploit the previous model by our recommendation algorithms, we convert the LDM to a WDG by collapsing all arcs between each two nodes into one single weighted edge. This step is applied as follows. Given the detailed LDM graph of a session  $s$ ,  $G_s = (N_s, A_s)$ , we compute its correspondent weighted graph  $G_s = (N_s, W_s)$ , where the arcs in  $W_s$  are of the form  $(n_i, n_j, u, w_s)$  and indicate that at least one action led the user from the query or document node  $n_i$  to  $n_j$ . The weight value  $w_s$  represents the probability that node  $n_j$  was relevant to the user for the given session. This value is either given explicitly by the user, or estimated by means of the implicit evidence obtained from the interactions of the user with that node, as follows:

$$w_s(n_i, n_j) = \begin{cases} 1, & \text{iff explicit relevance for } n_j \\ -1, & \text{iff explicit irrelevance for } n_j \\ lr(n_j) \in [0,1), & \text{otherwise (i. e. implicit relevance)} \end{cases}$$

In the case that there is only implicit evidence for a node  $n$ , the probability value is given by the *local relevance*  $lr(n)$ . This function returns a value between 0 and 1 which approximates the probability that node  $n$  was relevant to the user, based on the different interactions that the user had with the node. For instance, if the user opened a video and played it for the whole of its duration, this is taken as evidence that the video has a high chance of being relevant to the user [Fox et al. 2005]. Following this idea, and based on previous work on the impact of implicit feedback importance weights by Hopfgartner et al. [2007], the local relevance function is defined as  $lr(n) = 1 - \frac{1}{x(n)}$ , where  $x(n)$  is the total of added weights associated to each type of action leading to node  $n$ , which is defined as  $A_s(n) = \{(n_i, n_j, a, u, t) \in A_s | n_j = n\}, n \in N_s$ .

The weights associated to each type are natural positive values returned by a function  $f(a): A \rightarrow \mathbb{N}$  which maps action types to a number. These weights are higher for an action that is understood to give more evidence of relevance to the user. In this way,  $lr(n)$  is closer to 1 as more actions are observed that involve  $n$  and the higher the

associated weight given to each action type. In our weighting scheme some of the implicit actions are weighted nearly as highly as for explicit feedback. The accumulation of implicit relevance weights is thus calculated as  $x(n) = \sum_{a \in A_s(n)} f(a)$ . Table I shows an example of function  $f$ , based on the studied video retrieval system (see Section 3 for a further explanation on each action type). As was stated earlier, these weights are based on previous work on implicit feedback for video search [Hopfgartner et al. 2007]. Figure 2 shows an example of LDM and its corresponding WDG for a given session.

Table I. Values for the  $f$  function for the action types considered in the simulation. A higher value means a higher evidence for implicit relevance.

Action $a$	$f(a)$
Play (3 Sec)	3
View	10
Navigate	2
Browse	
R/L	
Tooltip	1

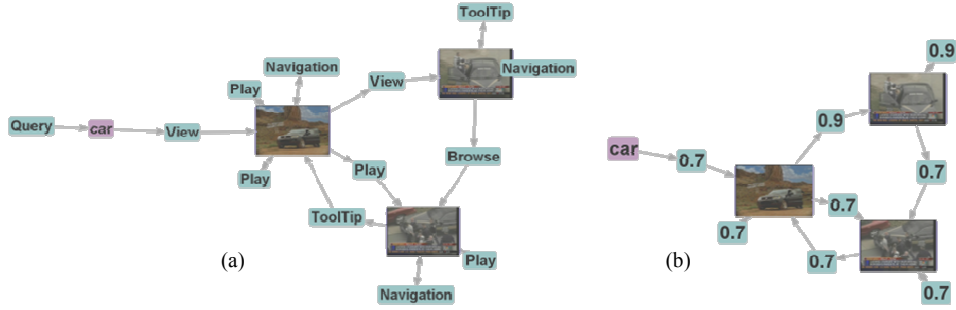


Fig. 2. Correspondence between the LDM (a) and WDG (b) models

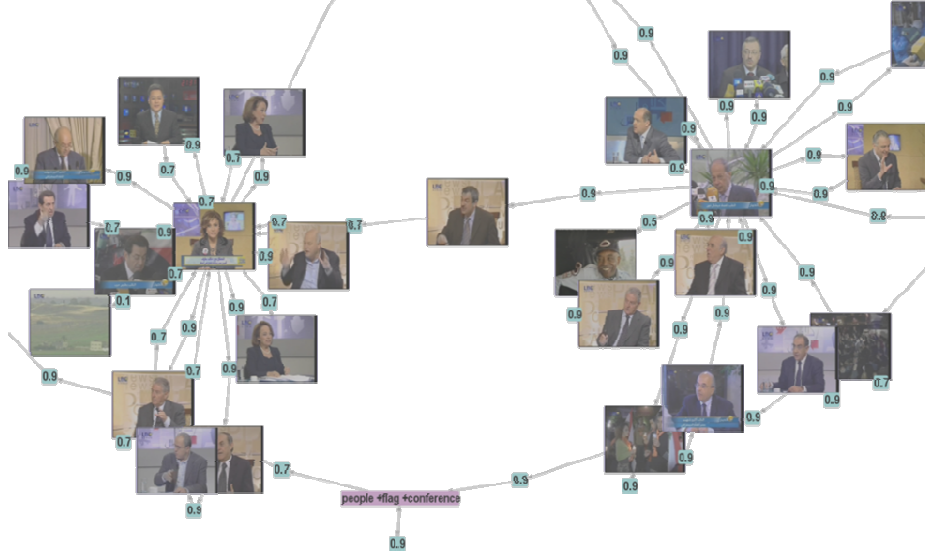


Fig. 3. Typical implicit pool graph structure, where some relevant nodes receive a large number of arcs

Similarly to the detailed LDM graph, the session-based WDGs can be aggregated into a single overall graph  $G = (N, W)$ , which we call the *implicit (relevance) pool*, as it

collects all the implicit relevance evidence of users from past sessions. The nodes of the implicit pool are all the nodes involved in any past interaction  $N = \bigcup_s N_s$ , whereas the weighted arcs in  $W$  are of the form  $(n_i, n_j, w)$ , where  $n_i, n_j \in N$  and  $w$  combines the probabilities of all the session-based values. In our approach we opted for simply averaging these probabilities, this is  $w = \frac{\sum_s w_s}{\#\{w_s\}}$ . Each link represents the overall implicit (or explicit, if available) relevance that all users whose actions led from node  $n_i$  to  $n_j$ , gave to node  $n_j$ . Figure 3 shows an example of implicit pool.

The presented recommendation approaches are based on the status of the current user session. As the user interacts with the system, a session WDG is built. The current user's session is thus represented by the ongoing session graph  $G_s = (N_s, W_s)$ , where in this case  $s$  is the current user's ongoing session. This graph is the basis of the recommendation algorithms presented next, which exploit the implicit pool in order to retrieve similar nodes that were somewhat relevant to other users.

#### 4.2 Graph-Based Recommendation Strategies

We have defined four recommendation strategies based on the implicit pool graph representation. The first two approaches, namely *query destination* and *random walk*, are adaptations of two important works in the state of the art [White et al. 2007; Craswell and Szummer 2007] to our implicit pool representation. Our adaptations extend these approaches with additional factors, thus a comparison to the original methods is included in this paper (see Section 6.1.2). Our third recommendation strategy, *interaction sequence*, is a novel approach that takes inspiration in state of the art techniques, but includes a new factor accounting for the temporal sequence of user interactions. The fourth recommendation approach, *neighbourhood*, was at first devised as a baseline approach, because of its simplicity, but results from some of the evaluation configurations indicates it can be a competitive strategy in some cases.

Table II. Comparison of graph-based recommendation strategies

Strategy	Clickthrough data	Additional implicit evidence	Implicit evidence aggregation	Search trails	Temporal sequence	Query similarity
White et al. [2007]	X			X		X
Query Destination (adap. White et al. [2007])	X	X	X	X		X
Craswell and Szummer [2007]	X		X			
Random Walk (adap. C & S [2007])	X	X	X		X	
Interaction sequence	X	X	X	X	X	
Neighbourhood	X	X	X			

Table II summarises the differences and commonalities between the four strategies and the original approaches presented by White et al. [2007] and Craswell and Szummer [2007]. The column *clickthrough data* indicates that the algorithm exploits the clickthrough information (queries and accessed documents) collected from users. *Additional implicit evidence* indicates that the strategy makes use of additional implicit evidence other than clickthrough data, such as viewing time or navigating through a video. The column *Aggregation of implicit evidence* indicates those approaches that aggregate in some way the implicit evidence from past users. Approaches that have a

mark in the *Search trails* column adopt the concept of search trails introduced by White et al. [2007]. *Temporal sequence* indicates that the ordered sequence of user interactions is exploited, while *query similarity* implies that the approach includes an additional content-based scoring factor: a similarity value between the current user query and the document. These methods are described next and evaluated and compared in Section 6.

**4.2.1 Query Destination.** This algorithm is adapted from the work of White et al. [2007] on query and search trails which we cited in precious sections. White et al. suggest that the last documents that a user visits right before inputting a query has a high relevancy for the current search session. We choose the query destination measure, which they proved to be best for explorative tasks (used in the evaluation process). The query destination value ranks by popularity the last accessed documents before each query, which are referred to as the query trails' destinations. White et al. define query trails as the interactions sequences that lead the user from one input query to the next, in which the destination is the last document that the user accessed after issuing the next query. The popularity value is based on the similarity of the document to the last query input by the user. We extend this popularity value by taking into consideration the link weights in the query trail, which proved to improve recommendation performance in our initial experiments [Vallet et al. 2008]. In our graph-based representation this recommendation approach is defined as follows:

$$qd(q, d) = S(d, q) \cdot \sum_{\substack{q \rightsquigarrow d_j \rightarrow d \rightarrow n_q \\ d_j, d \in D_s, n_q \in Q_s}} w(d_j, d)$$

where  $q \rightsquigarrow d_j \rightarrow d \rightarrow n_q$  denotes a search interaction trail from query  $q$  and the next user query  $n_q$ , and  $w(d_j, d)$  indicates the weight value associated to the last document of this search trail. We thus define a popularity value as the weight aggregation of all incoming arcs within the paths of the different historical query trails defined between  $q$  and  $d$ . We combine this popularity value with the original popularity value given by White et al.,  $S(d, q)$ , which we compute as the *tf.idf* similarity measure between document  $d$  and the last query  $q = \text{last}(Q_s)$  input by the user. This similarity value is a content-based feature, and depends on having associated text to the document. Note that though the arcs between documents in the WDG are essentially trail links, we do not limit these trails to accessed documents, but we extend them with more types of actions.

**4.2.2 Random Walk.** Craswell and Szummer [2007] exploit the clickthrough data with a random walk algorithm over the bipartite graph of queries and accessed documents. This recommendation approach uses the user's last input query as the seed of the random walk computation. There are two variations of this recommendation algorithm. In the first variation, the forward random walk approach, the final distribution value will assign, in theory, higher probability values on those nodes that past users found (implicitly) relevant after issuing the query. In the second variation, the backward random walk approach, these distribution values will be higher for those nodes that could be the "cause" of the last input query, i.e. those nodes that the user could have visited before issuing her last query.

In order to apply the random walk computation, the probability of going from node  $n_j$  to  $n_k$  is needed. We take a similar approach as Craswell and Szummer, by normalizing the transition probabilities between any two nodes  $n_j$  and  $n_k$ , and adding a self-transition probability. The transition probability from node  $n_j$  to node  $n_k$ , denoted as  $P(n_k | n_j)$ , is thus defined as:

$$P(n_k|n_j) = \begin{cases} (1-s) C_{jk} / \sum_i C_{ji} & \text{if } k \neq j \\ s & \text{when } k = j \end{cases}$$

where  $s$  is the probability of staying at the same node, and  $C_{ij}$  is the click count from node  $n_i$  to node  $n_j$ . We set  $s$  to 0.9, as Craswell and Szummer's experiments results point out that this value gives a slightly better performance, and we define the click count as the weight value in our implicit graph  $C_{ij} = w(n_i, n_j)$ , which not only takes into consideration the clickthrough data but also aggregates other sources of implicit relevance evidence. Note that the implicit pool is not bipartite and hence we can take into account link directionality, thus being able to exploit inter-document and document-query temporal transitions. Using these probability values, we can then define the recommendation approach as either the backwards random walk  $rw_B(q)$  or the forward random walk  $rw_F(q)$  applied over the implicit pool. The approach uses the last query of the current user session as the seed of the random walk computation, defined in our model as  $q \in \text{last}(Q_s)$ . Both random walks were computed using 11 computation steps, as Craswell and Szummer reported the best results with this setup.

**4.2.3 Interaction Sequence.** This approach gains inspiration from previous works in the state of the art [White et al. 2007; Craswell and Szummer 2007], but there are a number of key differences:

- Similar to Craswell and Szummer [2007], our approach represents queries and documents in the same graph. However, we represent the whole interaction sequence, unlike their approach, where the clicked documents are linked directly to the query node, forming a bipartite graph. As Craswell and Szummer pointed out, their approach simplifies the user's behaviour: "the user has limited memory, so forgets their previous location after each transition". On the other hand, our representation aims at recommending potentially important documents that are part of the interaction sequence.
- The main addition of this recommendation approach is the concept of users' interaction sequences. We define interaction sequence as an ordered list of interactions that a user performs over a search system during the retrieval process. This concept is analogous to the concept of search trails introduced by White et al. [2007]. However, we do not limit the possible recommended documents to those documents that are at the end of the search trail, but rather we recommend those documents that immediately follow these trails. The concept of search trail by White et al. is oriented to non-interactive text-based Web search engines. Although we believe that their hypothesis is consistent in the Web search scenario, we also believe that multimedia interactive search is better identified with an explorative task scenario, where the information need of the user is satisfied by a subset of relevant documents, rather than a single one [White and Roth, 2009]. Hence, our hypothesis is that during a multimedia search scenario the relevance of a document does not depend on when it was accessed during the search session.

The goal of this approach is to exploit previous similar interaction sequences from past users in order to find underlying order relations between video objects. Examples of this order relations are: 1) temporal relations, such as a list of videos that cover the timeline of a particular news event; 2) content relations, such as a video summary of a longer and more descriptive video or a video that is split in different pieces; or 3) community relations, such the user community's reaction video that refers to the one the user is viewing at the moment. The recommendation approach can then exploit these underlying

relations in order to recommend videos that are related to the current interaction sequence of the user. For instance, if a user has opened a video of news highlights, the recommendation could contain more in-depth stories that previous users found interesting to view next. Note that this concept is also somehow exploited by our adaptation of the random walk approach into the implicit pool, but the interaction sequence strategy focuses on scoring higher documents that follow the current interaction sequence of the user. This recommendation approach is defined as follows:

$$is(n, N_s) = \sum_{\substack{n_i \in N_s \\ p=n_i \rightsquigarrow n_j \rightarrow n \\ \text{length}(p) < L_{MAX}}} lr'(n_i) \cdot \xi^{\text{length}(p)-1} \cdot w(n_j, n)$$

where  $n_i \rightsquigarrow n_j$  denotes the existence of a path from  $n_i$  to  $n_j$  in the graph,  $n_j \rightarrow n$  means that  $n$  is adjacent to  $n_j$ , and the same notation is used as a shorthand to define  $p$  as any path between  $n_i$  and  $n$ , taking into consideration the link directionality.  $\text{length}(p)$  is counted as the number of arcs in path  $p$ , which must be less than a maximum length  $L_{MAX}$ . Nodes are weighted with the current session's local relevance  $lr'(n_i)$ . Finally,  $\xi$  is a distance decay factor, set to 0.8 in our experiments, a value which allowed weights to be propagated significantly up to six degrees of separation at most. This decay factor allows giving more importance to those documents that directly follow the interaction sequence, though if a document with intense interaction records occurs two or three steps away it may also be recommended.

**4.2.4 Query and Document Neighbourhood.** This method follows a neighbourhood-based strategy. A neighbourhood approach refers to a way of obtaining related nodes; we define the node neighbourhood of a given node  $n$ , as the nodes that are within a distance  $D_{MAX}$  of  $n$ , without taking the link directionality into consideration. These nodes are somehow related to  $n$  by the actions of the users, either because the users interacted with  $n$  after interacting with the neighbour nodes, or because they are the nodes the user interacted with after interacting with  $n$ . Note that this method has a more relaxed interpretation than the previous approaches of both concepts of search trails and interaction sequences. More formally, we define the node neighbourhood of a given node  $n$  as:

$$NH(n) = \{m \in N \mid \delta(n, m) < D_{MAX}\}$$

where  $\delta(n, m)$  is the shortest path distance between nodes  $n$  and  $m$ , and  $D_{MAX}$  is the maximum distance to consider a node as a neighbour. The best performing setting for this value, in our experiments, was  $D_{MAX} = 3$ .

Using the properties derived from the implicit graph, we can calculate an overall relevance value for a given node, as the aggregation of implicit relevance historically given by users to node  $n$ , when  $n$  was involved in the users' interactions. This value can be understood as some sort of global popularity value, based on the implicit evidence given by all users. Given all the incoming weighted arcs of  $n$ , defined by the subset  $W(n) = \{(n_i, n_j, w) \mid n_j = n\}$ , for  $n \in N$ , the overall relevance value for  $n$  is computed as:

$$or(n) = \sum_{w \in W(n)} w$$

Given the current session of a user and the implicit relevance pool, we define the neighbourhood recommendation value as:

$$nh(n, N_s) = \sum_{n_i \in N_s, n \in NH(n_i)} lr'(n_i) \cdot or(n)$$

where  $lr'(n_i)$  is the local relevance value computed for the current session of the user  $G_s$  as defined in Section 4.1.2, so that the relevance of the node to the current session is taken into consideration. We can then define the query neighbourhood recommendation value as the application of the neighbourhood recommendation value restricted to the queries related to the current user session:  $nh_q(n, N_s) = nh(n, Q_s)$ , where  $Q_s \subset N_s$  is the subset of query nodes in the implicit pool. Similarly, we can define the document neighbourhood recommendation value  $nh_d(n, N_s) = nh(n, D_s)$ , where  $D_s \subset N_s$  is the subset of document nodes in the ongoing user session, thus recommending nodes related to the documents involved in the user's interactions.

## 5. SIMULATION-BASED INTERACTIVE RETRIEVAL EVALUATION

As explained earlier, our evaluation approach requires usage information from past users, collected in the implicit pool graph, and the trace of the current (live) user's interaction, which serves as input to the recommendation algorithms to be evaluated. Following the notation of the previous section, these two elements are noted as  $G = (N, W)$  and  $G_s = (N_s, W_s)$ , respectively.

One option is to construct both the implicit pool graph  $G$  and the ongoing session graph  $G_s$  by monitoring a set of users interacting with a real search system. This evaluation approach was followed in our work reported in [Hopfgartner et al. 2008]. The main problem encountered in this evaluation approach was the need for a real system implementation and a large amount of human resources in order to conduct the human-centred evaluation. Because of this, we limited our evaluation to a single recommendation approach, as for each evaluated approach the experiments should have been repeated, with further user involvement.

Given the amount of resources required by this kind of evaluation, we consider the production of simulated instances of either  $G$  or  $G_s$ . One possibility is to exploit the implicit pool obtained during a user centred evaluation, which we denote by  $G_u$  henceforth, and simulate the ongoing user interaction  $G_s$ . This approach was explored in a previous work [Vallet et al. 2008], which has been extended and included in this paper (see Section 6.1). The main advantage of this approach is that a number of recommendation methods can be evaluated and compared without further user involvement. However, the main drawback is that the user-based implicit pool  $G_u$  used in this experiment is fixed. Thus, it only allows comparing the different recommendation approaches for a limited amount of search task representations and a single source of implicit information. In this section we will show how this implicit pool can be characterised, how it can be simulated in order to provide a more extensive evaluation of the recommendation approaches, and how the different variations of the implicit pool impact the performance of the presented strategies.

### 5.1 Implicit Graph Parameters

In the user-centred evaluation, the user-based implicit pool  $G_u$  was constructed by monitoring the interaction of 24 users [Hopfgartner et al. 2008] with an enhanced version of the video retrieval system described in Section 3. The enhancement consisted in the addition of a recommendation module. The participants' group consisted of 18 males and 6 females, mostly postgraduate students and research assistants, with an average age of 25.2 years and an advanced proficiency with English. Each of the users performed the same four explorative tasks selected from TRECVID 2006 [Over and Ianeva 2006], spending 15 minutes for each task. We decided to use those tasks that performed the worst in TRECVID, mostly due to their multifaceted and ambiguous nature, while still describing quite specific information needs, and therefore being the most challenging for current multimedia retrieval systems. The four tasks were:



1. Find shots with a view of one or more tall buildings (more than four stories) and the top story visible.
2. Find shots with one or more soldiers, police, or guards escorting a prisoner.
3. Find shots of a group including at least four people dressed in suits, seated, and with at least one flag.
4. Find shots of a greeting by at least one kiss on the cheek.

We therefore constructed the implicit pool  $G_u$ , which collected over 11K actions from all users, including  $\sim 1K$  textual queries. In order to include noisy data, we added the interaction information from two training tasks which users performed for ten minutes each. We then performed a statistical analysis over these interactions, with which we were able to calculate probability values of the user performing specific actions available in the retrieval system. Table III shows the probability values obtained from this user study.

Table III. Probability and normal distribution measures for observed action types

Action type	Probability	Action type	$\mu$	$\sigma$
$P(\text{Click} R)/P(\text{Click} \neg R)$	0.8/0.3	Navigation	0.5	2
$P(R \text{Click})$	0.07	Play interval (3 sec interval)	2	3
$P(\text{Tooltip} R)/P(\text{Tooltip} \neg R)$	0.8/0.4	Browsing near keyframe	0.25	1

The left column of Table III shows the probabilities associated to standalone actions, i.e. actions that can be triggered independently from others. The action types shown in the right column depend on previously clicking a document, as they cannot be performed without previously opening the document. The video retrieval system also allowed tooltipping a video search result, by leaving the mouse pointer on top of a result for longer than one second. This action shows a popup info box with the nearest keyframes neighbours and related text, with no need for opening the document. Once a document is opened (i.e. clicked), the user could navigate through its contents or browse to its neighbour shots (represented by their keyframes). The video play duration was also monitored, by triggering a play interval action every three seconds of video playing. The navigation, browsing and play interval actions can be characterised by a positive Normal Gaussian distribution, with a mean value of  $\mu$  and a typical deviation of  $\sigma$ , which indicate the number of times these actions are performed. We analysed the distribution of all types of actions over relevant and irrelevant documents and found a statistical difference on the click and tooltip actions. On the other hand, we did not find any significant difference on the playing, navigating, and browsing actions between not relevant and relevant documents. Hence, the probability values of clicking or tooltipping are defined as  $P(\text{Click}|R)/P(\text{Click}|\neg R)$  and  $P(\text{Tooltip}|R)/P(\text{Tooltip}|\neg R)$ , respectively, depending on the document being relevant or not.

The click type action is modelled by an additional parameter: the probability that a clicked document is relevant, denoted as  $P(R|\text{Click})$ . As the implicit pool interprets actions over a document as implicit relevance evidence, a value of  $P(R|\text{Click})$  close to 1 means that this evidence is close to the real relevance of the results. A value close to 0 would indicate that the user is mostly interacting with non-relevant documents, which could result in a low *quality* implicit pool, as it could be representing false evidence of relevance. Hence, this value gives an overall idea of the *quality* of the implicit graph that is being constructed, as the implicit pool is constructed from implicit evidence. In the forthcoming experiments we will define this parameter as  $Q(G)$ , so as to evaluate the effect the quality on implicit information over the performance of the recommendation strategies. Our hypothesis is that the better is the quality of the implicit pool, the better should be the output of the recommendation algorithms. Note that the quality value of the

user-based implicit graph  $Q(G_u)$ , created from our user experiment, is rather low: 0.07. This means that only 7% of user actions were involved with relevant documents. We hypothesise that this low value is due to the difficulty to evaluate a video document as relevant without opening or interacting with it. Even with this low value we achieved successful results in the user-centred evaluation [Hopfgartner et al. 2008]. The reason is that by the aggregation of the implicit evidence of multiple users we can mitigate the low quality value of the implicit pool.

Another relevant parameter to study is the available implicit information that the recommendation algorithm has access to. This amount of implicit information, seen as the number of actions, or the “size” of the implicit pool, can be characterised by the number of user actions from which the implicit pool is based. This value is equal to the number of graph link elements that belong to the LDM implicit pool  $G = (N, A)$ . We shall denote the size parameter by  $S(G)$ . Figure 4 shows the relation between the size of the implicit pool  $G_u$  (created from the user study) and the number of graph nodes (i.e. documents and queries) that this graph contains. The graph shows the delta variation of actions and nodes with respect to the first user as each user interaction information is added to the graph pool. The actions’ delta variation is the expected, as it increases constantly as more interactions are added to the implicit pool. This means that all users are providing a similar amount of actions to the implicit pool. The variation of nodes follows the  $x=y$  with the first users, but as more users are added into the implicit pool, the variation is below the expected. We hypothesise that actions are being concentrated on a subset of nodes, which can be those that are most relevant to users, and therefore can be exploited by the recommendation algorithm in some way, no matter the size of the implicit pool.

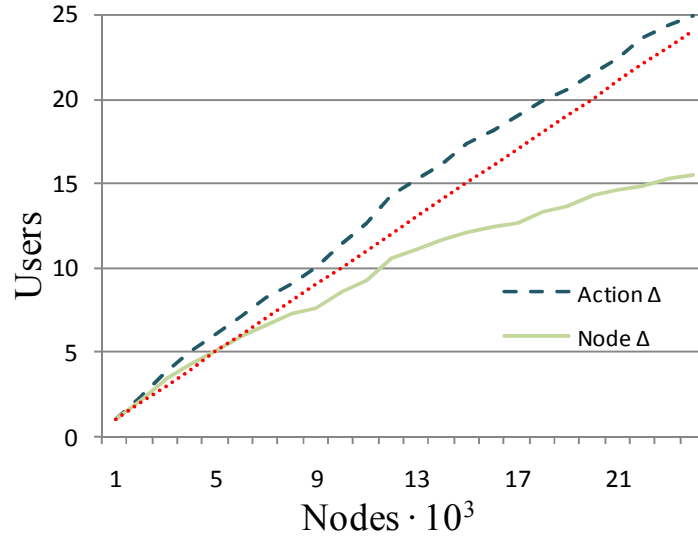


Fig. 4. Delta variations of actions and nodes in the implicit pool

## 5.2 Simulated Behaviour for Implicit Graph Construction

Once we have harvested the user-based implicit pool  $G_u$  from the user experiment, a natural next step would be to undertake new user studies in order to evaluate each recommendation strategy. However, we have four different recommendation strategies, six if we consider the variations of the neighbourhood and random walk approaches, which can make this evaluation too costly both in time and human resources. Instead of

this, we propose a simulation framework using the mined statistical data. Using this data, we can simulate users interacting with a video retrieval system and evaluate the performance of the recommendation approaches. Furthermore, we may analyse the performance of these approaches with implicit pools of different characteristics, and study issues such as the impact of the quality of the implicit evidence, or the scalability of our approaches.

The simulation's goal is to produce a simulated instance of a session graph  $G_s$ . In addition, the same as if they were produce from real users, simulated session implicit graphs can be used to construct a simulated implicit pool  $G = (N, W)$ . As explained in Section 4.1.2, this can be done by aggregating a set of simulated session implicit graphs  $G_s$  into the overall simulated implicit pool. A simulated session graph  $G_s$  is constructed by means of a number  $I$  of simulated interactions between the user and the retrieval system. We define an interaction as the execution of a user query and the possible interactions of the user with the retrieved results. Figure 5 depicts the simulation of an interaction  $i \in \{1, 2, \dots, I\}$  for a task  $T$ . Note that when  $i < I$  we can see  $G_s$  as a simulated ongoing session graph, as the simulated search session has not yet ended.

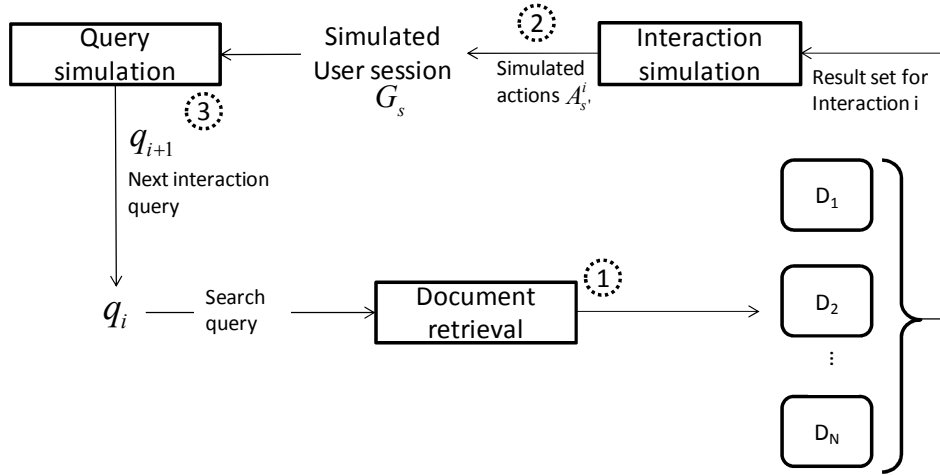


Fig. 5. Steps and related components for a user interaction simulation

The interaction simulation has the following steps:

- 1) Execute a simulated query, generated by the query simulation module in a previous interaction simulation (step 3). In case this is the first simulated interaction ( $i = 1$ ), the simulated query can be obtained by extracting the most relevant terms from a random set of relevant documents for the task  $T$  (which are available in the relevance assessment for the task).
- 2) Generate a set of simulated actions of the user on the returned result set. These actions are added to the simulated session graph. The actions are generated according to the probabilities shown in Section 5.1, with the following methodology:

```

for each item d in the top N query results
  if (d is relevant)
    //rand(P) returns TRUE if a random generated number between
    //[0,1] is lower than P
    tooltiped = rand (P(Tooltip|R))
    interacted = rand (P(R|Click)/P(Click|R))
  
```

```

        clicked = rand (P(Click|R))
    else
        tooltiped = rand (P(Tooltip|¬R))
        clicked = rand (P(Click|¬R))
        interacted = true
    if (tooltiped) //document tooltiped
        generate tooltip action
    else if (interacted) && (clicked) //document opened
        generate click action

        generate max(0,round(N(μbrowsing, σbrowsing))) browsing actions
        generate max(0,round(N(μnavigation, σnavigation))) navigation actions
        generate max(0,round(N(μplay, σplay))) playing actions

```

- 3) Generate the query for the next interaction. This query is obtained by extracting the three top most important terms from the documents involved in interactions in the previous step, by using query expansion techniques. The underlying idea is to simulate users refining their search queries with information from the documents that they interacted with.

The session graph  $G_s$  is thus created with the interaction information provided by the simulation process for task T. Each simulated session is generated with a value of  $I = 10$  simulated interactions, as this was the average interactions (i.e. queries) that the users performed for each task during the user evaluation study. With this simulated graph we can then construct a simulated version of the implicit pool  $G$ .

### 5.3 Simulated Behaviour for Interactive Retrieval Evaluation

Once an overall implicit graph is available (either from real users or simulated), we can apply a simulation-based technique in order to evaluate the different recommendation approaches. This evaluation methodology is basically a small extension of the one described in the previous section, by adding a recommendation approach and incorporating its output to the results interacted by the simulated user. It therefore also simulates a user interacting with the video retrieval system, but this time the user receives the recommendations from the evaluated algorithm, rather than solely the search system's results. In order to evaluate the recommendation algorithms, we made the following assumption: after a query is launched, users first inspect the top five recommended results (i.e. the output of the evaluated approach) before they continue inspecting the query result set. The simulated actions can then be generated following the same methodology as the previous section (step 2), but adding the top five recommended results on top of the result set. The new interaction simulation process is depicted in Figure 6.

Recommendation and query results are thus integrated into a single interaction result set (step 1a and 1b), in which the simulated actions are generated, just as in the previous simulation methodology. The simulated actions update the ongoing session graph  $G_s$  (step 2), which serves as input of the recommendation approach. Note that the interaction sequence and neighbourhood approaches can be updated as soon as any type of new implicit information is obtained. However, in order to evaluate the algorithms evenly, we choose to update the recommendation at the beginning of each interaction. The other input for the recommendation approach is the implicit pool  $G$ , which can be either created with real users or generated by the previous simulation approach (see Section 5.2).

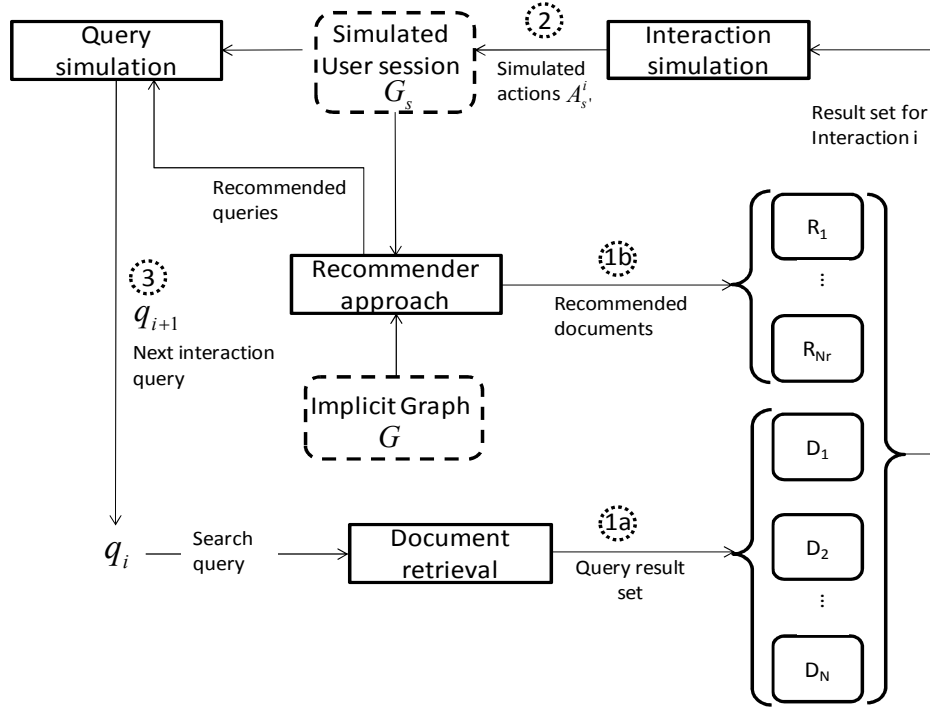


Fig. 6. Steps and related components for the simulated evaluation of a recommendation approach

The query generation strategy is also slightly changed by taking into account the possibility that the user chooses a recommended query (with a probability 0.6, accordingly to our user study data). The result set from each interaction step is used to evaluate the performance of the recommendation algorithm. This allows us to both evaluate the performance of the recommended documents (which are the top five results) and the recommended queries (which generate the query result set). The final result set for the simulated task process is generated by a rank-based merging of each  $i$  interaction result set (and again using 10 simulated interactions). This methodology draws from an evaluation approach used on a real system submitted to the TREVIDC interactive video retrieval task reported by Urban et al. [2006b].

## 6. EXPERIMENTAL RESULTS

The simulation results are discussed in this section. The TRECVID 2006 test collection is used [Over and Ianeva 2006]. We measure the performance of a recommendation algorithm with the Precision at N ( $P@N$ ) and Mean Average Precision (MAP) metrics over the final merged result set (see Section 5.3), using the relevance assessments provided by the test collection. These are the standard metrics used in the TRECVID 2006 interactive track.  $P@N$  is the ratio between the number of relevant documents in the first  $N$  retrieved documents and  $N$ . The  $P@N$  value focuses on the quality of the top results, with a lower consideration on the quality of the recall of the system. MAP is the mean Average Precision metric, which computes the average precision value for every cut of point given by the position of each relevant document found in the result set. This measure is normally used for a simple and convenient system's performance comparison. We show a  $P@N$  comparison when evaluating the recommendation approaches over a single type of implicit pool and an MAP comparison when the evaluating the impact of different variations of implicit graphs. Each metric is averaged over the number of tasks

involved in the simulation process. Furthermore, we average the results over 50 simulation runs, which resulted in statistically significant differences in most of the experiments (Wilcoxon,  $p < 0.05$ ), although in some cases the number of runs had to be increased. As a baseline system we opted for a simulation run with no recommendation whatsoever.

### 6.1 Experiments Over Real Implicit Pool Data

In this section we evaluate and compare the performance of the recommendation algorithms using the simulated task approach that makes use of a user generated implicit pool  $G_u$  (see section 5.1). The evaluation is only carried on the four tasks from which  $G_u$  was constructed.

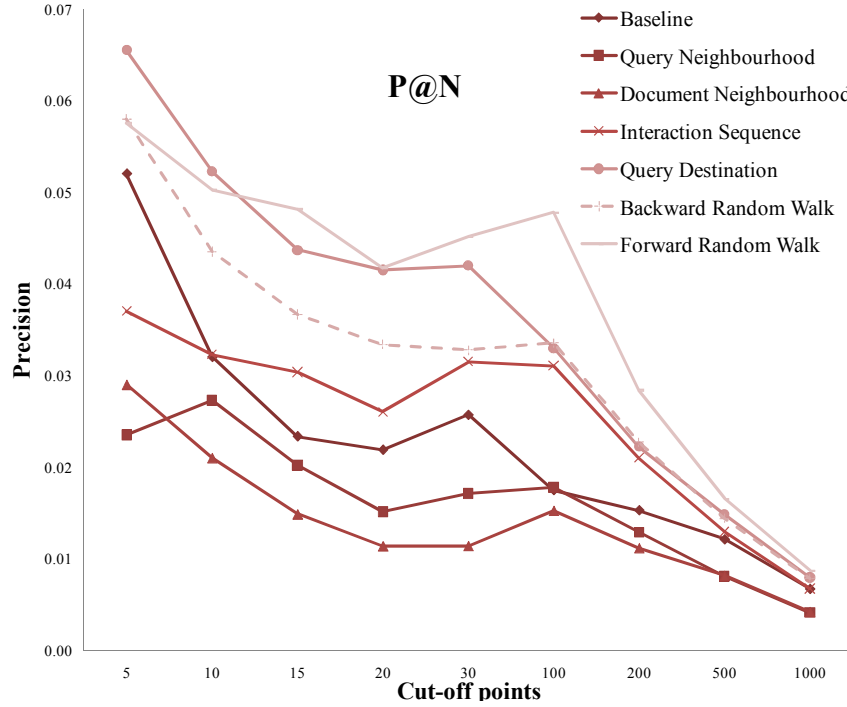


Fig. 7. Precision cut-off points for each recommendation strategy with user generated implicit pool

**6.1.1 Overall performance of recommendation approaches.** We here compare the performance of the recommendation approaches presented in section 4.2. The overall performance of each recommendation approach is shown in Figure 7. The recommendation strategy that overall appears to perform best is the query destination recommendation, followed by the random walk and interaction sequence approaches. One singular characteristic of the query destination approach is that the similarity between the last query and the recommended documents is also taken into consideration, apart from link weights. The performance of the query destination and interaction sequence algorithms does highlight the importance of exploiting the search and query trails similarities. The concept of interaction sequence, exploited by the interaction sequence approach and in a minor degree by the random walk strategy, also seems to be a common point of the best performing approaches. The backward random walk performs worse than forward random walk approach, although Craswell and Szummer [2007] reported the contrary. The neighbourhood based strategies perform below the baseline.

These approaches do not take into account sequences of interaction or search trails. This suggests that these factors, which are considered in the other recommendation approaches, may be important factors for an effective recommendation approach. An examination of the performance results for each of the four tasks showed that there were different algorithms that performed better in the tree first tasks: query destination in Task 1, interaction sequence in Task 2 and forward random walk in Task 3. Finally, no recommendation approach was able to outperform the baseline in Task 4. The reason was probably that users showed an erratic behaviour in this task, as they confessed a great difficulty on matching the semantics of the task at hand with the videos' textual metadata.

*6.1.2 Comparison with state of the art approaches.* As described previously, the query destination and the random walk recommendation strategies are adaptations of previous strategies proposed by White et al. [2007] and Craswell and Szummer [2007], respectively. We argued that our adaptations add additional factors from which the original approaches may benefit. In order to test this hypothesis, we make a direct comparison between the original versions and the proposed modifications.

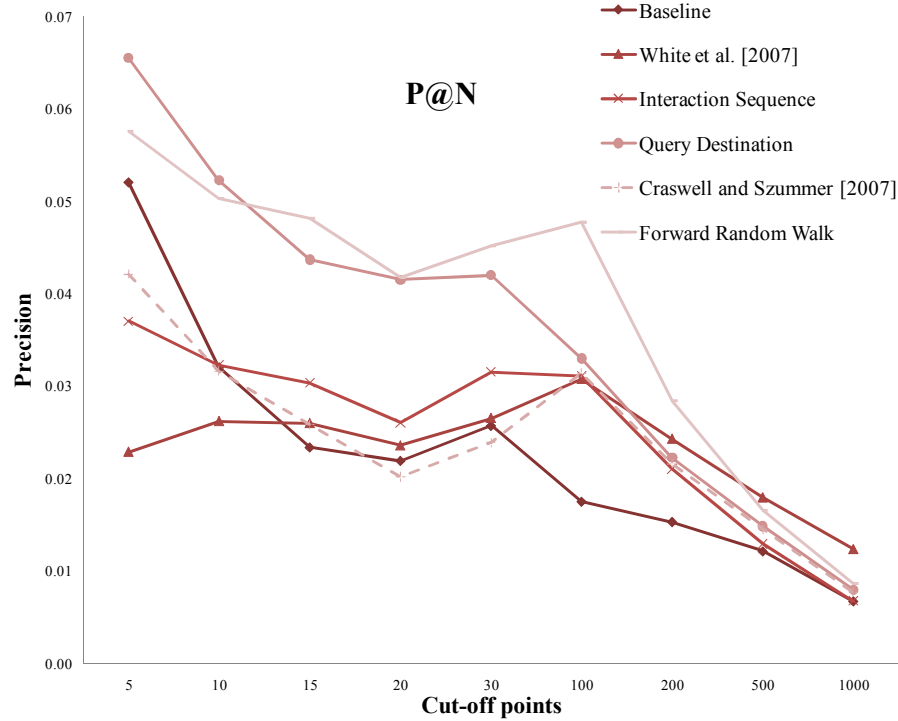


Fig. 8. Comparison between adapted and state of the art recommendation strategies

Figure 8 shows this analysis. In the comparison of the results by White et al. [2007] and the query destination approach, it is clear that our adaptation outperforms the original strategy. With respect to Craswell and Szummer's [2007] approach, we compare their best performing variation (forward random walk) with our adapted random walk approach. The results also show that our adapted strategy clearly outperforms the original approach. Additionally, our new proposed recommendation strategy, the interaction sequence, outperforms both original approaches. These results indicate that, in video retrieval, the additional evidence considered in the implicit pool gives an advantage to our recommendation strategies over the two state of the art methods, which only exploit

clickthrough data. Moreover, our adaptation of the approach by White et al. also includes the aggregation of past implicit evidence, represented in the implicit pool, which was not considered in the original approach. Our more elaborate representation of user interactions over a directed graph is also an advantage towards the bipartite representation of Craswell and Szummer's approach.

In order to further investigate the effect of considering additional implicit evidence apart from clickthrough data, we can compare the performance of our recommendation strategies over a user implicit pool constructed with clickthrough data and an implicit pool constructed with all the implicit evidence collected from users. The results indicate that all recommendation strategies, except for the backwards random walk, had a significant improvement when additional types of implicit evidence were considered. In the case of the backward random walk we notice a small decrease of improvement, however in the subsequent evaluations shown in this paper, this decrease of performance is not observed. This leads us to think that this was a circumstantial effect of this evaluation setup with only 4 evaluation tasks.

*6.1.3 Investigating variations over our recommendation strategies.* We here investigate possible variations of the 4 recommendation strategies. First, each recommendation strategy makes use of different scoring functions, with which they rank the final recommended content to the user. For instance, the neighbourhood-based algorithms use the *or* (overall relevance) scoring function for ranking (see Section 4.2.4); the query destination adds a content-based feature and calculates the similarity between the document and the query; and the random walk algorithms rank the documents by the final probability value. Based on this, we can create different ranking schemes, which vary the way these four original approaches rank the final recommendation, by combining their original ranking with a new ranking variation from other approaches.

Second, we test whether a more elaborate construction of the ongoing session graph  $G_s$  can have a significant impact on the observed results. The construction method described in Section 4.1.2 assigns a probability value to all the nodes (queries and documents) involved in a retrieval session, based on the amount and types of actions falling on a document or query. When building the implicit pool, this basic approach is suitable, as the session graphs are constructed offline, from ended retrieval sessions. However, there are runtime-based strategies that can be more suitable for a graph representing a live session [White et al. 2005]. We have thus tested an elaboration reflecting the drifting nature of themes or topics during a retrieval session, by introducing a time variable and a forgetting function that decreases the weights of observed items as the observation lags behind in time (as in the so-called ostensive model by Campbell and van Rijsbergen [1996]).

Following all the above considerations, different variations were tested, those who had significant results are listed below:

- Overall relevance. Re-rank recommendations with the *or* value of each node (see Section 4.2.4).
- Query similarity. Re-rank document recommendations with the tf-idf similarity between the document and the last executed query (see Section 4.2.1).
- Time decay. Apply a forgetting function in the generation of the session-based graph  $G_s$ .

We tested these different variations on the four original algorithms with the implicit pool  $G_u$  from the original 24 users and four topics evaluation set. Figure 9 shows a final comparison between the best performing algorithm variation for each original algorithm. Neither the interaction sequence nor the query destination algorithms obtained benefit from these variations.



Comparing these results with the initial ones shown in Figure 7, we observe that there is a boost of performance in the random walk and neighbourhood approaches. The best performing variations of both the query and the document neighbourhood approaches are using the overall relevance re-ranking and the ostensive model approach. The document neighbourhood approach is now equivalent in terms of performance to the query neighbourhood approach. However, both approaches still fall behind the baseline in this setup. Note that the neighbourhood approaches already use the overall relevance value in its score definition (see Section 4.2.4). Hence, we can suppose that boosting this *or* value has a benefit on their performance, although most of its performance boost comes from the use of the ostensive model for the ongoing session graph construction. The random walk approaches benefit from the overall relevance variation. In particular, the backward random walk has now a comparable performance to the query destination algorithm and a better performance on lower values of  $N$  than the forward random walk. A further examination on each of the four task's performance results shows that there are still the same top algorithms for each task, except for Task 3 where now the best performing algorithm is the backward random walk, instead of the forward approach. These experiments show that there is still room for improvement of the algorithms, and that the sophistication of the ranking procedures and the ongoing session graph construction can lead to better performing recommendation approaches. In the rest of our experiments we will use the best performing algorithms (variations or originals), although we will refer to them by their original denomination.

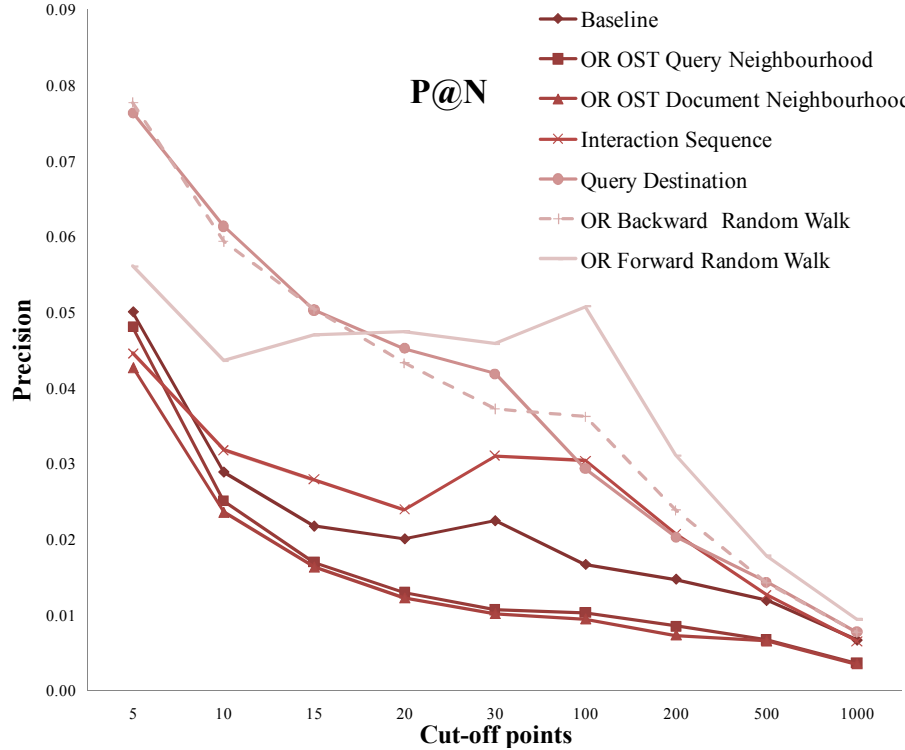


Fig. 9. Comparison between best performing strategy variations

## 6.2 Impact of Implicit Pool Attributes

Using the simulated implicit graph construction techniques presented in Section 5.2, we test the impact of two attributes of the implicit graph: the quality and the quantity of the implicit information. As mentioned before, the top algorithm variations were chosen for an extended evaluation over all 24 topics of TRECVID 2006. Rather than a direct performance comparison of the recommendation approaches, we are interested in analysing how these approaches adapt to implicit pools with different characteristics. The two tested parameters are:

- Quality of implicit information,  $Q(G)$ : This parameter gives an idea of the quality of the implicit information that the implicit pool represents. When a user interacts with a document, i.e. clicks a document result, the quality parameter gives the probability that this document is relevant to the user:  $P(R|Click)$ . With higher quality values, the implicit feedback is more accurate, as the percentage of relevant documents in the implicit graph increases. The parameter evaluates on the one hand the capability of a recommendation approach to give good results when poor implicit evidence is available and on the other hand if the recommendation can take advantage of high quality implicit evidence.
- Quantity of implicit or usage information,  $S(G)$ : This parameter represents the amount of information that the implicit pool contains for a specific topic. It is intended to evaluate how scalable is the recommendation approach when exploiting a greater amount of past implicit information. In order to be scalable, the algorithm has to be good on discarding noisy data, which comes from the increase of usage information in other unrelated topics and false implicit evidence.

*6.2.1 Effect of the Quality of Implicit Pool.* Analysing the data from our user study, we calculated that the quality value of the user-based implicit pool  $G_u$  was 0.07, i.e. when a user clicked on a video, there was a probability of 7% that that video was relevant. Based on this behaviour, we were interested to see how the variation of this quality influences the quality of our recommendations, as other types of retrieval systems can generate higher quality implicit pools. Image or text based systems would probably have a higher quality implicit pool, as users can more easily decide if a search result snippet is relevant to her task, before actually interacting with the search result. For instance, Joachims et al. [2007] report a quality value of ~80% for a textual retrieval system that offers text snippets of each result, whereas Craswell and Szummer [2007] report a quality value of ~75% in their image retrieval system.

In order to study the effect of the quality of the implicit pool, we simulated different implicit pools  $G$  with different levels of quality  $Q(G)$ . We used our implicit pool simulation approach (see Section 5.2) in order to create the implicit pools with different qualities, by varying the probability value  $P(R|Click)$ . We simulated 24 users performing each of the 24 TRECVID 2006 tasks, which produced implicit pools of an average size of  $S(G) \sim 65K$  actions. We used our simulation methodology in order to evaluate the recommendation approaches over these implicit pools (see Section 5.3). The plotted MAP of each approach shown in Figure 10 shows the results of this experiment.

As it was expected, all recommendation algorithms seem to benefit from an implicit pool with higher quality. However, there are two algorithms that seem to benefit the most from higher quality implicit pools: the query neighbourhood and the forward random walk. On the other hand, the interaction sequence and backward random walk seem to perform better at lower levels of quality, but do not perform as better as the previous two approaches on higher levels of quality. Overall, the interaction sequence approach seems to have the most stable performance over all the values of quality. The query neighbourhood approach, which performed worse than the baseline on lower levels of

quality, has a great leap on performance when more quality implicit pools are provided. It is interesting to see that this basic approach, which does not actually exploit the interaction sequences, has a similar performance, at high quality level of implicit pools, as the forward random walk approach. It is also worth pointing the differences on performance of the forward random walk versus the backward random walk, which were not appreciable at lower quality values, verifying the results obtained by Craswell and Szummer [2007].

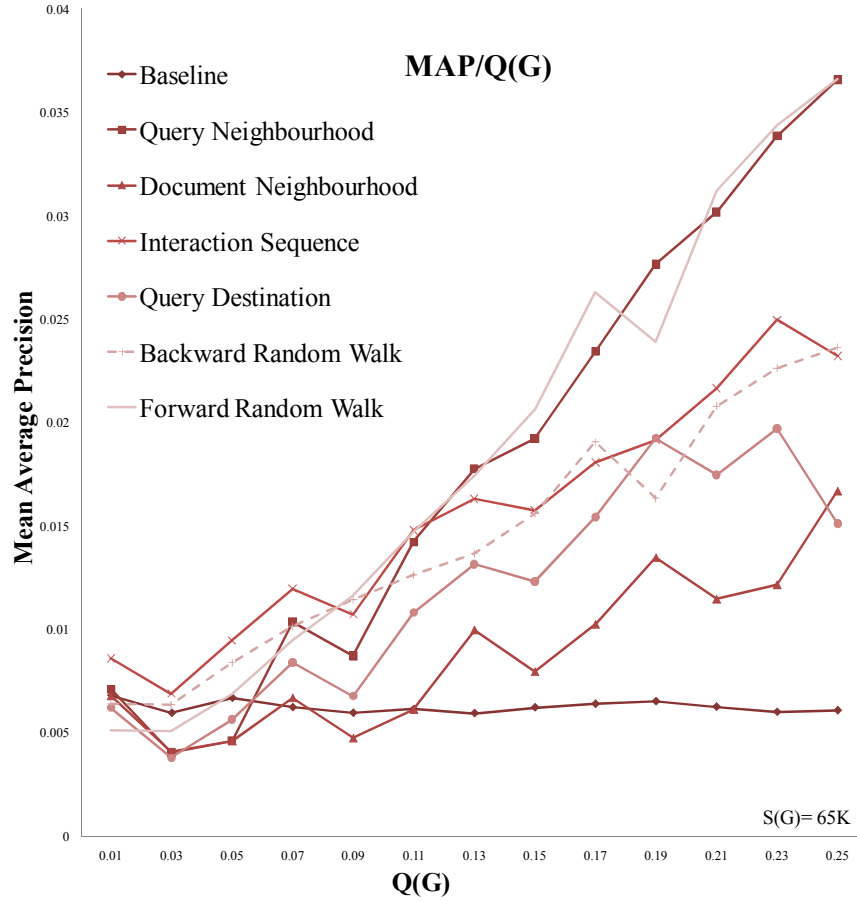


Fig. 10. MAP vs Implicit graph quality

**6.2.2 Effect of the Amount of Usage Information.** The motivation of this experiment is to test the scalability of our recommendation approaches. In principle, having more past usage information would lead to better recommendation results. But the increasing amount of noisy usage information collected from users performing unrelated search sessions can easily harm the performance of the recommendation approaches. Also, implicit pools with low amounts of usage information could simulate a document collection that has few interaction data. This could happen, for instance, in dynamic collections where there is fewer interaction data on novel documents.

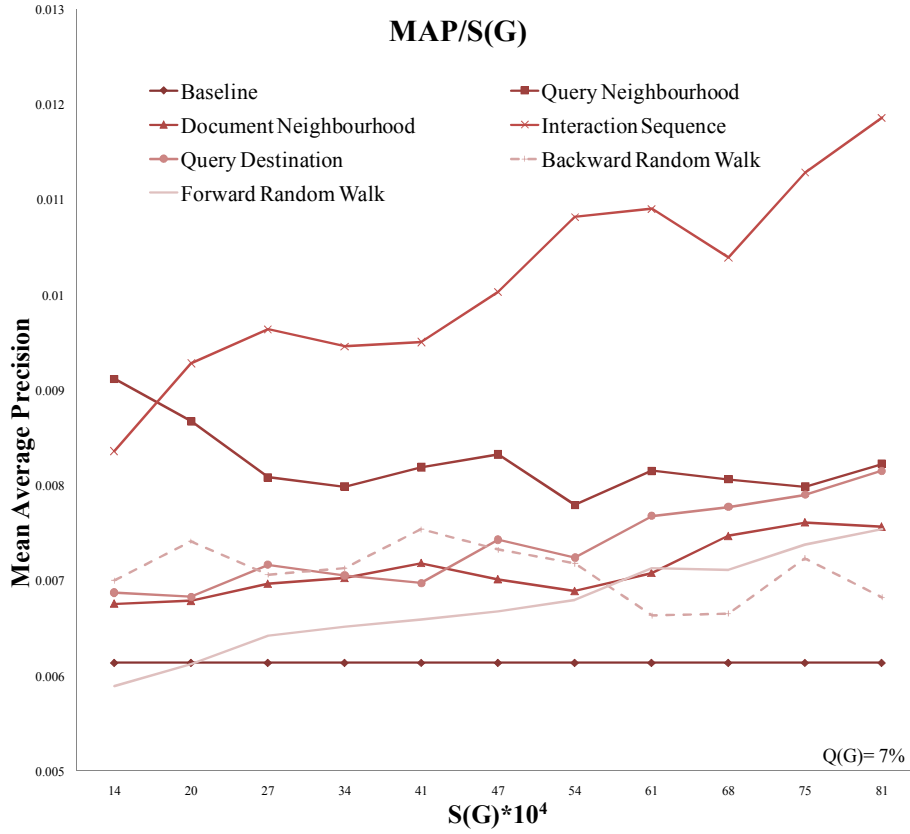


Fig. 11. MAP vs User data on implicit pool.

For each recommendation approach, we simulate up to 300 users executing all of the 24 TRECVID 2006 Tasks. In order to give an idea of the size of the simulated implicit pool, the graph created from the user study has a size of  $S(G_u) \sim 11K$  recorded actions, whereas we test up to a size of  $810K$  actions. The actions are added incrementally in each tested graph. In order to simulate realistic user behaviour, we use the quality value  $Q(G) = 7\%$  observed during our user study. Figure 11 shows the results of this experiment.

Analysing the simulation results, we can classify the algorithms on those that tend to show a constant performance, no matter the size of the implicit graph: backward random walk and document neighbourhood; those for which the performance tends to degrade with the number of users: query neighbourhood; and, finally, those which gain performance as more information is added to the implicit graph: interaction sequence, query destination and forward random walk. The interaction sequence approach proves to be the more scalable of all the strategies, better exploiting the amount of implicit information, without, apparently, being affected by a possible information overload or by an increase of noisy data.

## 7. CONCLUSIONS AND DISCUSSION

We have explored in this work the effective exploitation of community usage feedback to aid users in video retrieval tasks. We have introduced an integrated model which represents the past usage information as a graph-based implicit pool. The model results in an efficient and scalable way of representing this past information and, even more

important, supports a wide variety of recommendation strategies. The implicit pool representation provides a flexible ground that proves to facilitate the analysis of the diverse types of implicit actions that a video retrieval system can support.

In addition, an evaluation framework is introduced, the main goal of which is to facilitate the evaluation of recommendation strategies based on past usage information. This evaluation framework is based on simulated interactions, in which a user performing a given retrieval task with the evaluated system is simulated. We have presented a set of experimental results on the proposed recommendation strategies, starting from an implicit relevance pool generated by real users. Next, we have used the simulation approach to create our own implicit pools. We take advantage of the simulated approach to set the variables of the implicit feedback pool along different configurations, thus enabling a more comprehensive evaluation of the recommendation approaches. Two important variables of an implicit pool have been studied: quality and size. The former allows testing the impact of the reliability of the implicit evidence over the recommendation strategies; the latter allows testing their scalability. The simulation framework thus provides significantly extended means for the evaluation of the recommendation approaches. Though we have focused our research herein on video retrieval, the simulation framework is extendable to recommendation strategies in other types of retrieval systems, such as text or image-based.

### 7.1 Analysis of Experimental Results

The main conclusions drawn from the experiments in our evaluation framework are the following:

- **Recommendation strategies benefit from the implicit pool representation.**

The experiment over the user-generated implicit pool allowed a more direct comparison of the recommendation approaches. Two of the evaluated recommendation approaches, query destination and random walk, are adaptations of the approaches by White et al. [2007] and Craswell and Szummer [2007], respectively. A comparison between our adapted version and the original methods showed that the former perform significantly better than the latter, thus providing empiric evidence that the recommendation methods benefit from the implicit pool approach. Additionally, our proposed recommendation approach, interaction sequence, outperformed the state of the art approaches in the experiments.

- **Selection of a recommendation strategy depends largely on the characteristics of the gathered implicit evidence.**

Our experiments over simulated implicit pools validate our hypotheses on the impact of both the quality and scale of implicit feedback information on the performance of recommendation. Regarding quality of implicit evidence, although all recommendation strategies confirmed our hypothesis, there were differences in performance between approaches: on low quality pools the interaction sequence and backward random walk strategies have the best performance; on the other hand, the forward random walk and query neighbourhood take the best advantage of high quality pools. Regarding scale, the interaction sequence strategy, introduced in this work, was the only approach for which our hypothesis is clearly verified. All other approaches are at most scalable in terms of not degrading their performance. These results suggest that collaborative recommendation approaches based on implicit feedback largely depend on the quality and the amount of available usage information, and that these factors have to be considered when comparing different methods. As the characteristics of the available implicit information are tightly related to the characteristics of the retrieval system (e.g. the user interface, the

supported functionalities, the possibility of inputting relevance judgements without opening documents, and so forth), our study leads to the conclusion that these characteristics have to be taken into account in order to select an appropriate recommendation approach.

## 7.2 Comparison Between Recommendation Strategies

The reported experiments uncover the tradeoffs involved in the studied approaches. Table IV presents a summary of the performed experiments for different conditions of available implicit evidence.

Table IV. Performance values for different implicit pool characteristics. Table shows MAP values  $\times 10^{-2}$ . Values in bold show the best performing approach per row.

Strategy	Query Destination	Forward Random Walk	Backward Random Walk	Interaction Sequence	Document Neigh.	Query Neigh.
Low quality implicit evidence ( $Q(G) = 5\%$ )	0.5638	0.6874	0.8382	<b>0.9458</b>	0.4588	0.4612
High quality implicit evidence ( $Q(G) = 25\%$ )	1.511	<b>3.662</b>	2.363	2.322	1.669	3.657
Low amount implicit evidence ( $S(G) = 200K$ )	0.6874	0.5894	0.6998	<b>0.928</b>	0.6758	0.9116
High amount implicit evidence ( $S(G) = 800K$ )	0.8150	0.7536	0.6826	<b>1.1856</b>	0.7568	0.8224

The query-based variation of the neighbourhood method, which is one of the simplest approaches, has a very good performance over high quality implicit pools, although it does not scale well with pool size and has a poor performance over low quality implicit pools. The forward random walk approach shows a fairly good scalability, and very good performance over high quality implicit graphs, whereas the backward random walk displays opposite properties – it has a good performance over low quality implicit graphs but is not as scalable as the forward variant. The query destination approach is mediocre in terms of robustness to low implicit pool quality, but has a fair scalability. The interaction sequence approach, which is a novel strategy proposed in this work, appears to be the most stable of the compared approaches: it is the best performing method under low implicit information quality, and has a fair performance on higher quality levels. This approach builds on the hypothesis that the underlying order relationships between documents, as can be extracted from user interaction data, provide useful evidence for recommendation, by setting preference on those video objects that would appear to be in the user's way, based on the likely projections of his current interaction sequence. Furthermore, in terms of scalability it is the best recommendation strategy by far. Our analysis of the interaction information obtained from a group of users performing retrieval tasks with our experimental video retrieval system indicate that the quality of implicit feedback obtained from a video retrieval system is an order of magnitude lower than the implicit feedback obtained from image or text retrieval systems (7% compared to ~75-80%, see Section 6.2.1). This would account for the fact that, in video retrieval, the interaction sequence recommendation approach seems to fit better than other approaches available in the literature.

### 7.3 Discussion and Future Work

Our experiment setup assumes that users are performing similar search tasks, which can be thought as a rather strong assumption. The main constraint of having this assumption is the test collection. Task definitions in TRECVID are rarely related and make difficult to test our recommendation approaches with partially related search tasks. However, our scalability experiments did evaluate the effect of noisy data, mostly related to the simulated quality of implicit evidence, but also to partially overlapping tasks. Furthermore, in our related user study we carried a set of experiments in which we prove that these approaches behave similarly when evaluating their performance using related search tasks [Hopfgartner et al. 2008]. Moreover, TRECVID tasks imply that the user has to find as many relevant documents as possible, which may not fit all user search behaviour patterns. Our evaluation setup does simulate a user interacting with the retrieval system, and further refining their search queries. However further studies could indicate if our model can be generalised to other retrieval models, such as “berrypicking” models or exploratory search as defined by White and Roth [2009], where there is less emphasis on finding as many relevant documents as possible.

We have tested some variations of the recommendation methods, by combining some of the ranking scores used in these strategies, and by exploring a more complex construction approach for the ongoing user task representation. These experiments show how a more complex construction strategy can improve some of our recommendation approaches. A future study could consider other approaches of live representation of a search task [White et al. 2005], where their effect could be further analysed.

In both our user study and the simulated experiments we did not consider negative implicit feedback [Yang et al. 2007], although our model supports it. Negative feedback, which takes into account potential evidence on a document being irrelevant to the current user’s task, has to be handled with care, but could be used to further enhance the quality of the implicit feedback pool, enabling further recommendation performance improvements. Our simulation framework could also benefit from further refinements when modelling user search behaviour, such as taking into account the rank distance between observed documents [Dupret and Piwowarski 2008], or considering the adoption of more complex models of user interactions, such as the Bayesian framework presented by Guo et al. [2009].

### REFERENCES

- ARAMPATZIS, A. AND KAMPS, J. 2008. A study of query length. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, Singapore, ACM Press, New York, NY, USA, 811–812.
- ADOMAVICIUS, G. AND TUZHILIN, A. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17, 6, 734–749.
- BAUER, T. AND LEAKE, D. B. 2001. Real time user context modeling for information retrieval agents. In *Proceedings of the tenth international conference on Information and knowledge management*, Atlanta, Georgia, USA, November 2001, ACM Press, New York, NY, USA, 568–570.
- CAMPBELL, I. AND VAN RIJSBERGEN, C. J. 1996. The ostensive model of developing information needs. In *Proceedings of 2nd International Conference on Conceptions of Library Science*, Kobenhavn, DK, 251–268.
- CLEVERDON, C. W., MILLS, J., AND KEEN, M. 1966. Factors determining the performance of indexing systems.
- CRASWELL, N. AND SZUMMER, M. 2007. Random walks on the click graph. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, Amsterdam, The Netherlands, July 2007, CLARKE, C., L. A., FUHR, N., KANDO, N., KRAAIJ, W. AND DE VRIES, A. P., Eds. ACM Press, New York, NY, USA, 239–246.
- CUNNINGHAM, S. J. AND NICHOLS, D.M. 2008. How people find videos. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries*, Pittsburgh, Pennsylvania, July 2008, ACM Press New York, NY, USA, 201–210.

- DOU, Z., SONG, R., AND WEN, J. 2007. A large-scale evaluation and analysis of personalized search strategies. In *Proceedings of the 16th international World Wide Web conference*, Banff, Alberta, Canada, May 2007, ACM Press, New York, NY, USA, 572-581.
- DUPRET, G. E. AND PIWOWARSKI, B. 2008. A user browsing model to predict search engine click data from past observations. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 331-338, ACM Press, New York.
- FININ, T. W. GUMS: A General User Modeling Shell. 1989. In *User Models in Dialog Systems*, KOBSA, A. AND WAHLSTER, W., Eds. Springer-Verlag New York, Inc., New York, NY, USA, 411-430.
- FOLEY, C. AND SMEATON, A. F. Evaluation of Coordination Techniques in Synchronous Collaborative Information Retrieval. 2008. In *Proceedings of the First Collaborative Search Workshop*, Pittsburgh, Pennsylvania, July 2008.
- FOX, S., KARNAWAT, K., MYDLAND, M., DUMAIS, S., AND WHITE, T. 2005. Evaluating implicit measures to improve web search. *ACM Trans. Inf. Syst.*, 23, 2, 147-168.
- GUO, F., LIU, C., KANNAN, A., MINKA, T., TAYLOR, M., WANG, Y. M., AND FALOUTSOS, C. 2009. Click chain model in web search. In *Proceedings of the 18th international conference on World wide web*, pages 11-20, ACM Press, New York.
- JOHO, H., HANNAH, D., AND JOSE, J. 2009. Revisiting IR techniques for collaborative search strategies. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, Toulouse, France, Springer-Verlag, Berlin, Heidelberg, pages 66-77.
- HOPFGARTNER, F. AND JOSE, J. 2007. Evaluating the implicit feedback models for adaptive video retrieval. In *Proceedings of the international workshop on Workshop on multimedia information retrieval*, Augsburg, Bavaria, Germany, September 2007, ACM Press, New York, NY, USA, 323-331.
- HOPFGARTNER, F., URBAN, J., VILLA, R. AND JOSE, J. 2007. Simulated Testing of an Adaptive Multimedia Information Retrieval System. In *Proceedings of the International Workshop on Content-Based Multimedia Indexing*, Bourdeaux, France, June 2007, PINQUIER, J., Ed., IEEE Computer Society, 328-335.
- HOPFGARTNER, VALLET, D., HALVEY, M. AND JOSE, J. 2008. Search Trails using User Feedback to Improve Video Search. In *Proceedings of the ACM International Conference on Multimedia*, Vancouver, Canada, October, 2008, ACM Press, New York, NY, 339-348.
- JANSEN, B. J. AND SPINK, A. AND SARAJEVIC, T. 2000. Real life, real users, and real needs: A Study and analysis of user queries on the Web. *Inf. Process. Manage.* 36(2), 207-227.
- JOACHIMS, T., GRANKA, L., PAN, B., HEMBROOKE, H., RADLINSKI, F., AND GAY, G. 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. Inf. Syst.*, 25, 2, art. 7.
- KELLY, D. AND BELKIN, N. J. 2004. Display time as implicit feedback: understanding task effects. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, Sheffield, United Kingdom, July 2004, JÄRVELIN, K., ALLAN, J., BRUZA, P. AND SANDERSON, M., Eds. ACM Press: New York, NY, USA, 377-384.
- MEI, T., HUA, X-S, YANG, L., YANG, S-Q AND LI, S. 2007. VideoReach: An Online Video Recommendation System. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, Amsterdam, The Netherlands, July 2007, CLARKE, C., L. A, FUHR, N., KANDO, N., KRAAIJ, W. AND DE VRIES, A. P., Eds. ACM Press, New York, NY, USA, 767-768.
- NICHOLS, D.M. 1997. Implicit ratings and filtering. In *Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering*, November, 221-228.
- SALTON, G. AND BUCKLEY, C. 1988 Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.* 24, 5, 513-523
- SALTON, G. AND BUCKLEY, C. 1990. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41, 4, 288-297.
- SMEATON, A. AND OVER, P. AND KRAAIJ, W. 2006. Evaluation Campaigns and TRECVID. In MIR'06: Proceedings of the 8<sup>th</sup> International Workshop on Multimedia Information Retrieval, 321-330. ACM Press, New York
- SMEULDERS, A.W.M., WORRING, M., SANTINI, S., GUPTA, A., JAIN, R. Content-Based Image Retrieval at the End of the Early Years. *IEEE Trans. Patterns Anal. Mach. Intell.* 22, 12, 1349-1380, 2000.
- SUN, J.-T., ZENG, H.-J., LIU, H., LU, Y., AND CHEN, Z. 2005. Cubesvd: a novel approach to personalized web search. In *Proceedings of the 14th international conference on World Wide Web*, Chiba, Japan, May 2005, ACM Press, New York, NY, USA, 382-390.
- OVER, P. AND IANEVA, T. 2006. TRECVID 2006 Overview. In *Proceedings of the Text REtrieval Conference Video Workshop*, Gaithersburg, Maryland, November 2006.
- URBAN, J., JOSE J., AND VAN RIJSBERGEN, C. . 2006a. An adaptive technique for content-based image retrieval. *Multimedia Tools and Applications*, 31, 1, 1-28.



- URBAN, J., HILAIRE, X., HOPFGARTNER, F., VILLA, R., JOSE, J., SIRIPINYO, C., AND GOTOH, Y. 2006b. Glasgow University at TRECVID 2006. In *Proceedings of the Text REtrieval Conference Video Workshop*, Gaithersburg, Maryland, November 2006, 13–14.
- VALLET, D., HOPFGARTNER, F., AND JOSE, J. 2008. Use of implicit graph for recommending relevant videos: A simulated evaluation. In *Proceedings of the 30th European Conference on Information Retrieval*, Glasgow, Scotland, April 2008, MACDONALD, C., OUNIS, I., PLACHOURAS, V., RUTHVEN, I., AND WHITE, R. W., Eds. Springer-Verlag, Berlin, Heidelberg, 199–210.
- WHITE, R. 2004. Implicit Feedback for Interactive Information retrieval. PhD thesis.
- WHITE, R. W., BILENKO, M. AND CUCERZAN, S. 2007. Studying the use of popular destinations to enhance web search interaction. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, Amsterdam, The Netherlands, July 2007, CLARKE, C. L. A, FUHR, N., KANDO, N., KRAAIJ, W. AND DE VRIES, A. P., Eds. ACM Press, New York, NY, USA, 159–166.
- WHITE, R. W. AND ROTH, R. A. 2009. Exploratory search: Beyond the query-response paradigm. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 1–98.
- WHITE, R. W., RUTHVEN, I., JOSE, J. M., AND VAN RIJSBERGEN, C. J. 2005. Evaluating implicit feedback models using searcher simulations. *ACM Trans. Inf. Syst.*, 23, 3, 325–361.
- WHITE, R. W., JOSE, J. M., RIJSBERGEN, V. C. J. AND RUTHVEN, I. 2004. A Simulated Study of Implicit Feedback Models. In *Proceedings of the European Conference on Information Retrieval*, Sunderland, United Kingdom, April 2004, GOOS, G., HARTMANIS, J. AND VAN LEEUWEN, J., Eds. Springer-Verlag, Berlin, Heidelberg, 311–326.
- YANG, B., MEI, T., HUA, X.-S., YANG, L., YANG, S.-Q., AND LI, M. 2007. Online video recommendation based on multimodal fusion and relevance feedback. In: *Proceedings of the 6th ACM international conference on Image and video retrieval*, Amsterdam, The Netherlands, ACM Press, New York, NY, USA, 73–80.